

Tilburg University

The Complexity of Optimizing over a Simplex, Hypercube or Sphere

de Klerk, E.

Publication date:
2006

Document Version
Publisher's PDF, also known as Version of record

[Link to publication in Tilburg University Research Portal](#)

Citation for published version (APA):
de Klerk, E. (2006). *The Complexity of Optimizing over a Simplex, Hypercube or Sphere: A Short Survey*. (CentER Discussion Paper; Vol. 2006-85). Operations research.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



No. 2006–85

**THE COMPLEXITY OF OPTIMIZING OVER A SIMPLEX,
HYPERCUBE OR SPHERE: A SHORT SURVEY**

By Etienne de Klerk

September 2006

ISSN 0924-7815

The complexity of optimizing over a simplex, hypercube or sphere: a short survey

Etienne de Klerk
Department of Econometrics and Operations Research
Faculty of Economics and Business Studies
Tilburg University

Tilburg 5000 LE, The Netherlands
e.deklerk@uvt.nl

Abstract

We consider the computational complexity of optimizing various classes of continuous functions over a simplex, hypercube or sphere. These relatively simple optimization problems have many applications. We review known approximation results as well as negative (inapproximability) results from the recent literature.

Keywords: computational complexity, global optimization, linear and semidefinite programming, approximation algorithms

JEL code: C60.

1 Introduction

Consider the generic global optimization problem:

$$\underline{f} := \min \{f(x) : x \in K\}, \quad (1)$$

for some continuous, computable $f : K \mapsto \mathbb{R}$ and compact convex set $K \subset \mathbb{R}^n$, and let

$$\bar{f} := \max \{f(x) : x \in K\}.$$

In this short survey we will consider the computational complexity of computing or approximating \underline{f} (or \bar{f}) in the case where K is one of the following three sets:

- the standard (or unit) simplex:

$$\Delta_n := \left\{ x \in \mathbb{R}^n : \sum_{i=1}^n x_i = 1, x_i \geq 0 \right\},$$

- the unit hypercube $[0, 1]^n$,
- the unit sphere: $S_n := \{x \in \mathbb{R}^n : \|x\| = 1\}$.

Problem (1) has a surprising number of applications for these choices of K , and we only mention a few.

For the simplex, and quadratic f , the applications include finding maximum stable sets in graphs, portfolio optimization, testing matrix copositivity, game theory, and population dynamics problems (see the review paper by Bomze [4] and the references therein). A recent application is the estimation of crossing numbers in certain classes of graphs [13].

The following example is of optimization of a general non-polynomial f over the simplex, that occurs in multivariate interpolation, and in finite element methods (see [10]).

Example 1.1 *Given is a finite set of interpolation points $\Theta \subset \Delta_n$. Denote the fundamental Lagrange polynomial associated with an interpolation point $\theta \in \Theta$ by l_θ . In other words, for $x \in \Theta$:*

$$l_\theta(x) = \begin{cases} 1 & \text{if } x = \theta \\ 0 & \text{else.} \end{cases}$$

For a given $g : \mathbb{R}^n \rightarrow \mathbb{R}$, the associated Lagrange interpolant of g with respect to Θ is:

$$L_\Theta(g)(x) := \sum_{\theta \in \Theta} g(\theta) l_\theta(x).$$

Note that $L_\Theta(g)$ interpolates g at the points in Θ .

The associated Lebesgue constant is defined as:

$$\Lambda(\Theta) := \max_{x \in \Delta_n} \sum_{\theta \in \Theta} |l_\theta(x)|.$$

The Lebesgue constant is important in bounding the error of approximation, since one can show that

$$\|L_\Theta(g) - g\|_\infty \leq (1 + \Lambda(\Theta))\|g - p^*\|_\infty,$$

where the norm is the supremum norm on Δ_n , and p^ is the best possible polynomial approximation of g of the same degree as $L_\Theta(g)$.*

Thus, to compute the Lebesgue constant for Θ , we should maximize $f(x) = \sum_{\theta \in \Theta} |l_\theta(x)|$ over the simplex Δ_n .

For $K = [0, 1]^n$ and quadratic f , the examples include the maximum cut problem in graphs (see below). For general f , it includes many engineering design problems where simple upper and lower bounds on the variables are given, and no other constraints are present. These problems are sometimes referred to as ‘box constrained global optimization problems’.

For $K = S_n$ (the sphere), problem (1) becomes a minimal eigenvalue problem for a quadratic form $f(x) = x^T Q x$, by the Raleigh-Ritz theorem. For general quadratic functions it contains the trust region problem, that appears in many nonlinear programming algorithms as a sub-problem. For general forms (homogeneous polynomials), it contains the classical problem of deciding whether a given form is positive semidefinite.

2 Notions of approximation

Most of the optimization problems we will consider will be NP-hard, and we will therefore be interested in approximating the optimal values as well as possible in polynomial time.

One has to be careful when defining the notion of an approximation to an optimal solution of problem (1). The reason is that we usually do not know the range $\bar{f} - \underline{f}$ of function values on K in advance. If $\bar{f} - \underline{f}$ is small compared to a given ϵ , then it is not satisfactory to only compute some $x \in K$ with the property that $f(x) - \underline{f} < \epsilon$. It is therefore better to find an $x \in K$ such that $f(x) - \underline{f} < \epsilon(\bar{f} - \underline{f})$, since we then know that $f(x)$ belongs to the ϵ fraction of lowest function values.

The next definition is based on this idea, and has been used by several authors, including Ausiello, d'Atri and Protasi [1], Bellare and Rogaway [3], Bomze and De Klerk [5], De Klerk, Parrilo and Laurent [12], Nesterov et al. [18], and Vavasis [20].

Definition 2.1 *A value ψ_ϵ is called a $(1 - \epsilon)$ -approximation of \underline{f} for a given $\epsilon \in [0, 1]$ if*

$$|\psi_\epsilon - \underline{f}| \leq \epsilon(\bar{f} - \underline{f}). \quad (2)$$

The approximation is called implementable if $\psi_\epsilon = f(x_\epsilon)$ for some $x_\epsilon \in K$.

If we replace condition (2) by the condition

$$|\psi_\epsilon - \underline{f}| \leq \epsilon, \quad (3)$$

then we speak of a $(1 - \epsilon)$ -approximation of \underline{f} in the weak sense.

The following definitions are essentially from De Klerk, Laurent and Parrilo [12], and are consistent with the corresponding definitions in combinatorial optimization.

Definition 2.2 (Polynomial time approximation algorithm) *Fix $\epsilon > 0$ and a class of continuous, computable functions on K , say \mathcal{F} . An algorithm \mathcal{A} is called a polynomial time $(1 - \epsilon)$ -approximation algorithm for problem (1) for the function class \mathcal{F} , if the following holds:*

1. *For any instance $f \in \mathcal{F}$, \mathcal{A} computes an $x_\epsilon \in K$ such that $f(x_\epsilon)$ is an implementable $(1 - \epsilon)$ -approximation of \underline{f} ;*
2. *the number of operations required for the computation of x_ϵ is bounded by a polynomial in n and in the bit size required to represent f .*

If, in addition, the number of operations to compute x_ϵ is bounded by a polynomial in $1/\epsilon$, the algorithm \mathcal{A} is called a strongly polynomial time $(1 - \epsilon)$ -approximation algorithm for problem (1) for the function class \mathcal{F} .

For example, if f is a polynomial, the bit size required to represent f may be taken as the sum of the bit sizes of its coefficients.

Definition 2.3 (PTAS/FPTAS) *If, for a given function class \mathcal{F} , problem (1) has a polynomial time $(1 - \epsilon)$ -approximation algorithm for each $\epsilon \in (0, 1]$, we say that problem (1) allows a polynomial time approximation scheme (PTAS) for the function class \mathcal{F} .*

In case of a strongly polynomial time $(1 - \epsilon)$ -approximation algorithm for each $\epsilon \in (0, 1]$, we speak of a fully polynomial time approximation scheme (FPTAS).

These definitions can be adapted in an obvious way for maximization problems, or if the approximations are in the weak sense of (3).

3 Inapproximability results

We first review negative approximation results for problem (1). We will see that, in a well-defined sense, optimization over the hypercube is much harder than over the simplex, while the complexity of optimization over a sphere is somewhere in between.

3.1 The case of the simplex

If $K = \Delta_n$, then computing \underline{f} is an NP-hard problem, already for quadratic polynomials, as it contains the maximum stable set problem as a special case. Indeed, let G be a graph with adjacency matrix A and let I denote the identity matrix; then the maximum size $\alpha(G)$ of a stable set in G can be expressed as

$$\frac{1}{\alpha(G)} = \min_{x \in \Delta_{|V|}} x^T (I + A)x$$

by a theorem of Motzkin and Straus [14]. Moreover, this problem cannot have a FPTAS, unless all problems in NP can be solved in randomized polynomial time; this is due to the following inapproximability result for the maximum stable set problem by Håstad [8].

Theorem 3.1 (Håstad) *Unless $NP = ZPP$, one cannot approximate $\alpha(G)$ to within a factor $|V|^{(1-\epsilon)}$ for any $\epsilon > 0$.*

Corollary 3.1 *Unless $P = ZPP$, there is no FPTAS for problem (1) for the class of quadratic functions and $K = \Delta_n$.*

3.2 The case of the hypercube

If $K = [0, 1]^m$ and f quadratic, then problem (1) contains the maximum cut problem in graphs as a special case. Indeed, for a graph $G = (V, E)$ with Laplacian matrix L , the size of the maximum cut is given by (see [7, 15]):

$$|\text{maximum cut}| = \max_{x \in [-1, 1]^{|V|}} \frac{1}{4} x^T L x = \max_{x \in [0, 1]^{|V|}} \frac{1}{4} (2x - e)^T L (2x - e), \quad (4)$$

where e is the all-ones vector. The maximum cut problem does not allow a PTAS, as the following theorem shows.

Theorem 3.2 (Håstad [9]) *Unless $P=NP$, there is no polynomial time 16/17-approximation algorithm for the maximum cut problem (4).*

It follows that problem (1) does not allow a PTAS for any class of functions that includes the quadratic polynomials if $K = [0, 1]^n$.

A related negative result is due to Bellare and Rogaway [3], who proved that if $P \neq NP$ and $\epsilon \in (0, 1/3)$, there is no polynomial time $(1 - \epsilon)$ -approximation algorithm in the weak sense for the problem of minimizing a polynomial of total degree $d \geq 2$ over all sets of the form $K = \{x \in [0, 1]^n \mid Ax \leq b\}$.

3.3 The case of the sphere

Nesterov [16] showed that maximizing a cubic form (homogeneous polynomial) on the unit sphere is an NP-hard problem, using a reduction from the maximum stable set problem.

Theorem 3.3 (Nesterov) *Consider a graph $G = (V, E)$ with stability number $\alpha(G)$. One has*

$$\sqrt{1 - \frac{1}{\alpha(G)}} = 3\sqrt{3} \max_{\|x\|^2 + \|y\|^2 = 1} \sum_{\substack{i < j \\ \{i, j\} \notin E}} y_{ij} x_i x_j.$$

Note that this indeed involves maximizing a (square free) form of degree 3 in the variables x and y over the unit sphere. Also note that the number of variables is polynomial in $|V|$, since the x variables correspond to the vertices of G , and the y variables correspond to the edges of the complement of G .

In view of the inapproximability result for the maximum stable set problem in Theorem 3.1, we have the following corollary.

Corollary 3.2 *Unless $NP=ZPP$, there is no FPTAS for minimizing square free degree 3 forms over the unit sphere.*

4 Approximation results

4.1 The case of the simplex

We now consider the complexity of (approximately) solving problem (1) for $K = \Delta_n$.

Easy cases

Problem (1) can be solved in polynomial time or allows a FPTAS for the following classes of functions:

- f is concave; in this case the global minimum of f is attained at one of the n extreme points of Δ_n ;

- f is convex and self-concordant with polynomial time computable gradient and Hessian. In this case the theory of interior point algorithms of Nesterov and Nemirovski [17] provides a FPTAS for problem (1). This remains true if K is the unit hypercube or unit ball.

Results for polynomial f

Bomze and De Klerk [5] showed that, for $K = \Delta_n$ and f quadratic, problem (1) allows a PTAS. One of the PTAS algorithms that they considered is particularly simple: it evaluates f on the regular grid

$$\Delta(n, m) := \{x \in \Delta_n : mx \in \mathbb{N}_0^n\},$$

and returns the lowest value. In other words, it computes the value

$$f_{\Delta(n, m)} := \min_{x \in \Delta(n, m)} f(x).$$

Note that $|\Delta(n, m)| = \binom{n+m}{m}$ which is a polynomial in n for fixed m . Bomze and De Klerk [5] showed the following.

Theorem 4.1 (Bomze, De Klerk) *Let f be quadratic. One has*

$$f_{\Delta(n, m)} - \underline{f} \leq \frac{1}{m}(\bar{f} - \underline{f}) \quad (5)$$

for any $m \geq 1$.

Corollary 4.1 (Bomze, De Klerk) *There exists a PTAS for minimizing quadratic polynomials over the unit simplex.*

This PTAS result was extended to polynomials of fixed degree by De Klerk, Laurent, and Parrilo [12]. Earlier, related results were obtained by Faybusovich [6].

Theorem 4.2 (De Klerk, Laurent, and Parrilo [12]) *Let $f(x)$ be a form of degree d and $r \geq 0$ an integer. Then,*

$$f_{\Delta(n, r+d)} - \underline{f} \leq (1 - w_r(d)) \binom{2d-1}{d} d^d (\bar{f} - \underline{f}), \quad (6)$$

where

$$w_r(d) := \frac{(r+d)!}{r!(r+d)^d} = \prod_{i=1}^{d-1} \left(1 - \frac{i}{r+d}\right). \quad (7)$$

One can verify that

$$1 - \binom{d}{2} \frac{1}{r+d} \leq w_r(d) \leq 1,$$

which implies that $\lim_{r \rightarrow \infty} w_r(d) = 1$.

Corollary 4.2 (De Klerk, Laurent, and Parrilo [12]) *Fix $d \in \mathbb{N}$. There exists a PTAS for minimizing forms of degree d over the unit simplex.*

There also exist more sophisticated (and practical) PTAS algorithms for minimizing forms of fixed degree over the simplex that employ linear or semidefinite programming. For example, the authors of [12] consider

$$f_{\min,LP}^{(r)} := \max \lambda \quad \text{s.t. the polynomial } \left(\sum_{i=1}^n x_i \right)^r \left(f(x) - \lambda \left(\sum_{i=1}^n x_i \right)^d \right) \text{ has nonnegative coefficients } (r = 0, 1, \dots)$$

Note these bounds may be computed using linear programming, and this computation may be performed in polynomial time when r and d are fixed. Moreover, it is shown in [12] that

$$0 \leq \underline{f} - f_{\min,LP}^{(r)} \leq \left(\frac{1}{w_r(d)} - 1 \right) \binom{2d-1}{d} d^d (\bar{f} - \underline{f}), \quad (r = 0, 1, \dots),$$

where $w_r(d)$ was defined in (7). Thus the values $f_{\min,LP}^{(r)}$ also provide a PTAS, since

$$\lim_{r \rightarrow \infty} w_r(d) = 1.$$

One may also define semidefinite programming (SDP) based bounds that are at least as strong as the LP ones:

$$f_{\min,SDP}^{(r)} := \max \lambda \quad \text{s.t. the polynomial } \left(\sum_{i=1}^n x_i^2 \right)^r \left(f(x \circ x) - \lambda \left(\sum_{i=1}^n x_i^2 \right)^d \right) \text{ is a sum of squares of polynomials } (r = 0, 1, \dots),$$

where ‘ \circ ’ is the component-wise (Hadamard) product. Note that $\underline{f} \geq f_{\min,SDP}^{(r)} \geq f_{\min,LP}^{(r)}$.

Results for non-polynomial f

Recently, De Klerk, Elabwabi, and Den Hertog [11] derived approximation results for (not necessarily polynomial) functions that meet a Lipschitz condition of given order.

Once again, the underlying algorithm is simply the evaluation of f on a suitable regular grid.

Before defining this class of functions, recall that the modulus of continuity of f on a compact convex set K is defined by

$$\omega(f, \delta) := \max_{\substack{x, y \in K \\ \|x - y\| \leq \delta}} |f(x) - f(y)|.$$

We now define the class $\text{Lip}_L(\alpha)$ of functions that meet the Lipschitz condition of given order $\alpha > 0$ with respect to a given constant $L > 0$:

$$\text{Lip}_L(\alpha) := \{f \in C(\Delta_n) : \omega(f, \delta) \leq \delta^\alpha L\}. \quad (8)$$

This condition is also called a Hölder continuity condition. (Some authors reserve the term ‘Lipschitz’ for the case $\alpha = 1$.)

Theorem 4.3 (De Klerk, Elabwabi, Den Hertog) *Let $\epsilon > 0$, $\alpha > 0$, and $L > 0$ be given and assume $f \in \text{Lip}_L(\alpha)$ (see (8)). Moreover, assume $f(x)$ can be computed in time polynomial in the bit size of x and the bit size needed to represent f .*

Then, for

$$m = \left\lceil \left(\frac{2L}{\epsilon} \right)^{\frac{1}{2\alpha}} \right\rceil,$$

one has

$$f_{\Delta(n,m)} - \underline{f} \leq \epsilon,$$

where $f_{\Delta(n,m)} := \min_{x \in \Delta(n,m)} f(x)$ can be computed in polynomial time.

This theorem implies a PTAS *in the weak sense* for minimizing computable functions from the class $\text{Lip}_L(\alpha)$ over Δ_n , for fixed L and α .

Note that Theorem 4.3 does not imply Corollary 4.2. De Klerk, Elabwabi, and Den Hertog [11] also identified a further class of functions that allow a PTAS. This class includes the polynomials of fixed degree, and is defined in terms of suitable bounds on the higher order derivatives of f . The interested reader is referred to [11] for more details.

It is still an open question to completely classify the classes of functions that allow a PTAS.

4.2 The case of the hypercube

For the maximum cut problem there is a celebrated polynomial time 0.878-approximation algorithm due to Goemans and Williamson [7], who suggested the following semidefinite programming (SDP) relaxation of the maximum cut problem (4):

$$|\text{maximum cut}| \leq \overline{OPT} := \max_X \left\{ \frac{1}{4} \text{trace}(LX) \mid \text{diag}(X) = e, X \succeq 0 \right\}, \quad (9)$$

where e is the all-ones vector, and $X \succeq 0$ means that X is a symmetric positive semidefinite matrix.

Goemans and Williamson [7] also devised a randomized rounding scheme that uses the optimal solution of (9) to generate cuts in the graph. Their algorithm produces a cut of cardinality at least $0.878 \overline{OPT} \geq 0.878 |\text{maximum cut}|$.

Related approximation results for quadratic optimization over a hypercube were given by Nesterov [15] and Nesterov *et al.* [18]. In particular, the Nesterov showed the following.

Theorem 4.4 (Nesterov [15]) *There exists a (randomized) polynomial time $2/\pi$ approximation algorithm for the problem of maximizing a convex quadratic function over $[0, 1]^n$.*

Notice that the objective function in the maximum cut problem (4) is convex quadratic, since the Laplacian matrix of a graph is always positive semidefinite. Thus the theorem by Nesterov covers a larger class of problems than maximum cut, but the $2/\pi$ constant is significantly lower than the 0.878 obtained by Goemans and Williamson.

4.3 The case of the sphere

The complexity of optimization over the sphere is still relatively poorly understood, compared to the simplex or hypercube. To be more precise, there still is a big gap between approximation and inapproximability results.

Easy case: quadratic optimization over the unit ball

Consider the problem of minimizing a quadratic function $f(x) = x^T Bx + 2a^T x + \alpha$ over the unit ball $\|x\| \leq 1$. By the S-procedure of Yakubovich (see e.g. [19] and the references therein) this problem may be rewritten as a semidefinite program as follows.

$$\min_{\|x\| \leq 1} f(x) = \max_{\beta, \tau \geq 0} \beta$$

subject to

$$\begin{bmatrix} B & a \\ a^T & \alpha - \beta \end{bmatrix} \succeq \tau \begin{bmatrix} -I & 0 \\ 0^T & 1 \end{bmatrix}.$$

Results for polynomial objective functions

The result in Corollary 4.2 implies that there exists a PTAS for minimixing *even* forms of fixed degree on the unit sphere. (Recall that a form is called even if all exponents are even.)

Recently, Barvinok [2] has proved another partial result: one may derive a randomized PTAS for maximizing a form on the sphere for a special class of forms called (δ, N) -focused forms.

Definition 4.1 (Barvinok) *Assume f is a form of degree d . Fix a number $0 < \delta \leq 1$ and a positive integer $N > d$. We say that $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is (δ, N) -focused if there exist N non-zero vectors c_1, \dots, c_N in \mathbb{R}^n such that*

- *for every pair (i, j) the cosine of the angle between c_i and c_j is at least δ ;*
- *the form f can be written as a non-negative linear combination*

$$f(x) = \sum_{\substack{I \subset \{1, \dots, N\} \\ |I|=d}} \alpha_I \prod_{i \in I} c_i^T x,$$

where the α_I 's are nonnegative scalars.

Theorem 4.5 (Barvinok) *There exists an absolute constant $\gamma > 0$ with the following property. For any $\delta > 0$, for any positive integer N , for any (δ, N) -focused form $f : \mathbb{R}^n \rightarrow \mathbb{R}$ of degree d , for any $\epsilon > 0$, and any positive integer*

$$k \geq \gamma \epsilon^{-2} \delta^{-2} \ln(N + 2),$$

the inequality

$$(1 - \epsilon)^{d/2} \max_{x \in S_n \cap L} f(x) \leq \left(\frac{k}{n}\right)^{d/2} \bar{f} \leq (1 - \epsilon)^{-d/2} \max_{x \in S_n \cap L} f(x)$$

holds with probability at least $2/3$ for a random k -dimensional subspace $L \subset \mathbb{R}^n$.

One may solve the problem $\max_{x \in S_n \cap L} f(x)$ in polynomial time using techniques from computational algebraic geometry, since L is of fixed dimension, and therefore the number of variables in the resulting optimization problem is in fact fixed. Thus we obtain the following corollary.

Corollary 4.3 (Barvinok) *Fix $\delta > 0$, $N \in \mathbb{N}$, and $d \in \mathbb{N}$. There exists a (randomized) PTAS for minimizing (δ, N) -focused forms of degree d over the unit sphere.*

It is still an open question whether this result can be extended to all forms of fixed degree.

5 Conclusion and discussion

Approximation algorithms have been studied extensively for combinatorial optimization problems, but have not received the same attention for NP-hard continuous optimization problems. Indeed, most of the results described in this survey were obtained in the last decade.

There is also not much computational experience yet with approximation algorithms for nonlinear programming. The only significant exception so far is for semidefinite programming relaxations for quadratic optimization on the simplex or the hypercube.

It is therefore the hope of this author that this relatively young research area will attract both theoretically and computationally minded researchers.

References

- [1] G. Ausiello, A. D'Atri, and M. Protasi: Structure preserving reductions among convex optimization problems, *Journal of Computer and System Sciences* **21** (1980), 136-153.

- [2] A. Barvinok: Integration and optimization of multivariate polynomials by restriction onto a random subspace, *Foundations of Computational Mathematics*, to appear.
- [3] M. Bellare and P. Rogaway: The complexity of approximating a nonlinear program, *Mathematical Programming* **69** (1995), 429-441.
- [4] I.M. Bomze: Regularity versus degeneracy in dynamics, games, and optimization: a unified approach to different aspects, *SIAM Review* **44**(3) (2002), 394-414.
- [5] I. Bomze and E. De Klerk: Solving standard quadratic optimization problems via linear, semidefinite and copositive programming, *Journal of Global Optimization* **24**(2) (2002), 163-185.
- [6] L. Faybusovich: Global optimization of homogeneous polynomials on the simplex and on the sphere, In C. Floudas and P. Pardalos (eds.), *Frontiers in Global Optimization*, (Kluwer Academic Publishers, 2004), 109-121.
- [7] M.X. Goemans and D.P. Williamson: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, *Journal of the ACM* **42**(6) (1995), 1115-1145.
- [8] J. Håstad: Clique is hard to approximate within $|V^{1-\epsilon}|$, *Acta Mathematica* **182** (1999), 105-142.
- [9] J. Håstad: Some optimal inapproximability results, *Journal of the ACM* **48** (2001), 798-859.
- [10] J.S. Hesthaven: From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex, *SIAM Journal on Numerical Analysis* **35**(2) (1998), 655-676.
- [11] E. de Klerk, D. den Hertog, and G. Elabwabi: On the complexity of optimization over the standard simplex, *European Journal of Operational Research*, to appear.
- [12] E. de Klerk, M. Laurent, P. Parrilo: A PTAS for the minimization of polynomials of fixed degree over the simplex, *Theoretical Computer Science*, to appear.
- [13] E. de Klerk, J. Maharry, D.V. Pasechnik, B. Richter, and G. Salazar: Improved bounds for the crossing numbers of $K_{m,n}$ and K_n . *SIAM Journal on Discrete Mathematics* **20** (2006), 189-202.
- [14] T.S. Motzkin and E.G. Straus: Maxima for graphs and a new proof of a theorem of Túrán, *Canadian Journal of Mathematics* **17** (1965), 533-540.
- [15] Yu. Nesterov: Semidefinite relaxation and nonconvex quadratic optimization, *Optimization Methods and Software* **9** (1998), 141-160.

- [16] Yu. Nesterov: Random walk in a simplex and quadratic optimization over convex polytopes. *CORE Discussion Paper 2003/71*, CORE-UCL (2003).
- [17] Yu. Nesterov and A.S. Nemirovski: *Interior point polynomial algorithms in convex programming*, (SIAM Studies in Applied Mathematics, **13**, SIAM, Philadelphia, USA, 1994).
- [18] Yu. Nesterov, H. Wolkowicz, and Y. Ye: Semidefinite programming relaxations of nonconvex quadratic optimization, In H. Wolkowicz, R. Saigal, and L. Vandenberghe (eds.), *Handbook of semidefinite programming*, (Kluwer Academic Publishers, Norwell, MA, 2000), 361-419.
- [19] B. T. Polyak: Convexity of quadratic transformations and its use in control and optimization. *Journal of Optimization Theory and Applications*, 99 (1998), pp. 553–583.
- [20] S. Vavasis: Approximation algorithms for concave quadratic programming, In C.A. Floudas and P. Pardalos (eds.), *Recent Advances in Global Optimization*, (Princeton University Press, 1992)), 3-18.